



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/765,488	01/18/2001	Robert L. Bradee	C543.12-1	9879

7590

05/07/2004

David R. Fairbairn
Kinney & Lange, P.A.
THE KINNEY & LANGE BUILDING
312 South Third Street
Minneapolis, MN 55415

EXAMINER

WU, ALLEN S

ART UNIT	PAPER NUMBER
2135	3

DATE MAILED: 05/07/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/765,488

Applicant(s)

BRADEE, ROBERT L.

Examiner

Allen S. Wu

Art Unit

2135

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 18 January 2001.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-26 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-26 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 17 January 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 3.
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. Claims 1-9 are rejected under 35 U.S.C. 103(a) as being unpatentable over Moriconi et al (hereinafter Moriconi), US Patent 6,158,010, in view of Andrews, US Patent 6,574,736.

As per claim 1, Moriconi discloses a method of providing computer application security (see for example, abstract), comprising: identifying secured resources within a software applications (see for example; col 6 ln 4-15); grouping secured resources into user roles in a data store (see for example col 6 ln 64-col 7 ln 3); creating a plurality of surrogate identifiers in a data store, each surrogate identifier being associated with one user role (see for example; col 7 ln 66-col 8 ln 6). Each local role is mapped to a global role. Furthermore, in computer programming, an identifier must be assigned to identify such a role. One of ordinary skill in the art at the time of the applicant's invention would have realized the surrogate identifier needing to be present in order to identify the global role in the computer programming art. Moriconi further discloses associating users with user roles, each user being associated with one user role; (see for example col 6 ln 64-col 7 ln 3) and determining access rights to the secured resources for each user (see for example col 8 ln 23-32) according to an identifier (see for example; subject col 8 ln 23-27).

As for the determining according to a corresponding surrogate identifier without disclosing the corresponding surrogate identifier to the user, the

corresponding surrogate identifier being associated with one user role of the user. Moriconi further discloses that access rights are determined according to a request consisting of a privilege, an object, and a subject (see for example; col 8 ln 23-27) and that a subject comprises of a user role (see for example; col 6 ln 64-67). One of ordinary skill in the art at the time of the applicant's invention would have realized the combination of using a surrogate (global role) identifier as the subject for making such access requests. Access determination is well known in the art to provide the advantages of simplifying processing of user identification and access credential mappings by reducing the amount of identifiers to be mapped.

As for determining without disclosing the corresponding surrogate identifier, Moriconi disclose a systems administrator for performing security policies (see for example, col 11 ln 50-65), but is silent on the means of associating users to user roles. Andrews further teaches the system administrator for associating users with a user role (see for example; col 13 ln 41-57), and each user role being associated with a surrogate identifier (see for example; role id, col 11 ln 9-15). One of ordinary skill in the art at the time of the applicant's invention would have realized that in such a scheme, only the system administrator has access to the surrogate identifier and that the identifier is only used to map access credentials relating to the identifier in the system and not disclosed to the user. Both Moriconi and Andrews discloses a means of authorizing a user to access resources. It would have been obvious to one of

ordinary skill in the art at the time of the applicant's invention to combine the teachings of Andrews within the system of Moriconi because it would have provided a secure and organized means of associating users to a user role and made the process of authorizing a user simpler.

As per claim 2, Moriconi further discloses identifying functions within the software application to be secured, the identified functions being secured resources (see for example, col 6 ln 4-14); and invoking a security call before permitting access to the secured resources (see for example; col 10 ln 42-51).

As per claim 3, Moriconi further discloses installing an embedded module in the software application to capture the security call (see for example; API, col 10 ln 42-51).

As per claim 4, Moriconi further discloses establishing in the data store links to each secured resources (see for example; col 6 ln 23-31 and col 8 ln 64-col 9 ln 4). One of ordinary skill in the art at the time of the applicant's invention would have realized the use of links as mapping to such resources in the database for performing criteria checking. Moriconi further discloses selecting links corresponding to related secured resources (see for example col 9 ln 33-44); grouping the selected links into user roles (see for example; col 6 ln 64-col 7 ln 4); and storing the user roles in the data store (see for example; col 9 ln 1-9).

As per claim 5, Moriconi further discloses establishing in the data store links to each secured resources (see for example; col 6 ln 23-31 and col 8 ln 64-col 9 ln 4). One of ordinary skill in the art at the time of the applicant's invention would have realized the use of links as mapping to such resources in the database for performing criteria checking. Moriconi further discloses selecting links corresponding to related secured resources (see for example col 9 ln 33-44); grouping the selected links into privilege sets (see for example; col 6 ln 51-63 and col 7 ln 55-58) grouping the privilege sets and links into user roles (see for example; col 6 ln 64-col 7 ln 4); and storing the user roles in the data store (see for example; col 9 ln 1-9).

As per claim 6, Moriconi further discloses establishing in the data store links to each secured resources (see for example; col 6 ln 23-31 and col 8 ln 64-col 9 ln 4). One of ordinary skill in the art at the time of the applicant's invention would have realized the use of links as mapping to such resources in the database for performing criteria checking. Moriconi further discloses selecting links corresponding to related secured resources (see for example col 9 ln 33-44); grouping the selected links into privilege sets (see for example; col 6 ln 51-63 and col 7 ln 55-58); grouping privilege sets and links into job functions (see for example col 6 ln 40-43); grouping job functions, privilege sets, and links into user

roles (see for example; col 6 ln 64-col 7 ln 4); and storing the user roles in the data store (see for example; col 9 ln 1-9).

As per claim 7, Moriconi further discloses associating a surrogate identifier with one user role in the data store (see for example col 6 ln 64-66). As for replicating each surrogate identifier in a data store of a security provider, Moriconi further discloses the use of multiple security providers for increased performance (see for example; col 11 ln 9-17). One of ordinary skill in the art at the time of the applicant's invention would have realized the need to replicate each surrogate identifier in each data store of a security provider so that proper user identification can be maintained in each provider.

As per claim 8, Moriconi discloses the claimed limitations as described above. Moriconi further discloses associating users with one user role (see for example col 6 ln 64-col 5 ln 3). As for the user role being undisclosed to the user, Andrews discloses a means of assigning users to roles (see for example col 12 ln 38-67) by administrators (see for example; col 13 ln 41-47). Andrews further discloses creating a list of user identifiers corresponding to existing users (see for example; col 12 ln 38-67 and fig 8); selecting user identifiers from the list (see for example col 13 ln 41-49 and fig 8); and storing selected user identifiers in the data store (see for example; fig 8). Andrews further teaches the system administrator for associating users with a user role (see for example; col 13 ln

41-57), and each user role being associated with a surrogate identifier (see for example; role id, col 11 ln 9-15). One of ordinary skill in the art at the time of the applicant's invention would have realized that in such a scheme, only the system administrator has access to the surrogate identifier and that the identifier is only used to map access credentials relating to the identifier in the system and not disclosed to the user. As for such a list on a security provider, one of ordinary skill in the art would have realized that such a mapping of user identifiers to roles on the security provider of Moriconi is necessary for such user authorization pertaining to the user role.

As per claim 9, Moriconi further discloses authenticating the user as a valid user (see for example; col 4 ln 1-18), authorizing the user to access one of the secured resources (see for example col 8 ln 23-32).

2. Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over Moriconi et al (hereinafter Moriconi), US Patent 6,158,010, in view of Andrews, US Patent 6,574,736, in view of Boitana, US Patent 6,158,010, and further in view of Serbinis et al (hereinafter Serbinis), US Patent 6,314,425.

As per claim 10, Moriconi further discloses invoking programmatically, an embedded component within the software application when a secured resource is accessed (see for example col 11 ln 7-17); passing a resource name identifying the secured resource through the embedded component to a platform

coordinator (see for example; col 8 ln 23-31 and col 10 ln 64-66), retrieving the user surrogate identifier associated with the user role from the data store (see for example; col 8 ln 23-31); evaluating automatically the surrogate identifier against the data store of the security provider (see for example, col 8 ln 23-31), determining automatically permissions associated with the surrogate identifier on the security provider (see for example; col 8 ln 23-31 and col 13 ln 14-55); and relaying the access rights to the software application through the embedded component (see for example; col 8 ln 25-31 and col 7 ln 7-17). The relaying of access rights is inherent to pass authorization to the application requesting authorization credentials.

Moriconi and Andrews are silent on the means of authenticating the user, but discloses that the system is compliant with any authentication scheme (see for example; col 4 ln 1-18). Boitana further discloses retrieving an identifier and a security provider name from the user (see for example; col 6 ln 4-25), passing the identifier and the security provider name to a security broker (see for example; operating system interface, col 6 ln 3-25); relaying the identifier to a security provider associated with the security provider name for authentication (see for example; operating system security software; col 6 ln 17-25); evaluating automatically the identifier against a data store of the security provider (see for example; RACF col 6 ln 17-35). Operating system security software, such as RACF, is well known in the art to use such an identifier as a lookup to authenticate the user. Furthermore, the returning of the authentication request to

the security broker is inherent for the determination of authentication by the security broker (operating system software). Both Moriconi-Andrews and Boitana disclose a means of authenticating and authorizing a user to secured resources. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Boitana within the teachings of Moriconi-Andrews because it would have provided a means of authenticating a user across different security providers, thus increasing security of the system by allowing only authenticated users to be authorized to such secured resources.

As for retrieving the user role associated with the user identifier from the data store. One of ordinary skill in the art at the time of the applicant's invention would have realized that such retrieving means is necessary when performing user role mapping to the access credentials of the combination of Moriconi-Andrews-Boitana combination.

As for the use of an authentication token with a time stamp in a cache of the security broker and permissions token with a time stamp on the platform coordinator, Andrews further discloses the use of tokens that are associated with a user access request (see for example, col 22 ln 10-25). Serbinis further discloses the use of authentication/permissions token with a timestamp to access resources (services) (see for example; col 20 ln 54-col 21 ln 27). Access tokens are well known in the art for the use of ensuring that only users that are validated are allowed to use resources (see for example; Serbinis, col 20 ln 54-27). Both Serbinis and the Moriconi-Andrews-Boitana combination disclose a means of

user authentication and authorization to resources. It would have been obvious to combine the teachings of Serbinis within the Moriconi-Andrews-Boitana combination because it would have provided increased security through extra validation of an authenticated and authorized user of such a resource.

As for storing such the tokens in the cache of the security broker and platform coordinator, caches are well known in the art to be used for temporary storage of data for processing. One of ordinary skill in the art at the time of the applicant's invention would have realized that such tokens must be temporarily stored in order to perform processing of the tokens and that caches are widely known in the art to be used for such temporary storage of information.

As for relaying the permissions token to the platform coordinator, Serbinis further discloses relaying the token to a client for future authentication (see for example, col 21 ln 5-30). When using such tokens in the Moriconi-Andrews-Boitana combination, one of ordinary skill in the art at the time of the applicant's invention would have realized the relaying of permissions token in place of the authentication response.

3. Claims 11-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Moriconi et al (hereinafter Moriconi), US Patent 6,158,010, in view of Andrews, US Patent 6,574,736, in view of Wobber, US Patent 5,235,642.

As per claim 11, Moriconi further discloses invoking programmatically, an embedded component within the software application when a secured resource

is accessed (see for example col 11 ln 7-17); passing a resource name identifying the secured resource through the embedded component to a platform coordinator (see for example; col 8 ln 23-31 and col 10 ln 64-66), retrieving the user surrogate identifier associated with the user role from the data store (see for example; col 8 ln 23-31); passing the surrogate identifier and the resource name from the security broker to the security provider (see for example; col 8 ln 23-31); evaluating automatically the surrogate identifier against the data store of the security provider (see for example, col 8 ln 23-31), determining automatically permissions associated with the surrogate identifier on the security provider (see for example; col 8 ln 23-31 and col 13 ln 14-55); and relaying the access rights to the software application through the embedded component (see for example; col 8 ln 25-31 and col 7 ln 7-17). The relaying of access rights is inherent to pass authorization to the application requesting authorization credentials.

Moriconi-Andrew is silent on the specific authentication means of user. Wobber et al discloses an authentication means including retrieving a token (record) from a cache (see for example; col 6 ln 7-22); passing the token to another server (see for example; col 7 ln 20-50); and comparing the authentication token against the cache on the server to identify a matching authentication token (see for example; col 7 ln 21-65). As for the token being associated in the cache with the surrogate identifier, one of ordinary skill in the art at the time of the applicant's invention would have realized that the use of tokens with the Moriconi-Andrew combination would have resulted in the need to

map the token to the proper surrogate identifier so that the appropriate user and corresponding access rights are identified. Moriconi-Andrew and Wobber disclose a means of user authentication and authorization to resources. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Wobber within the Moriconi-Andrew combination because it would have provided a means of granting authentication and access to resources in a quicker means through the use of tokens in a cache (see for example; Wobber, col 2 ln 5-23).

As for the use of an authentication token with a time stamp in a cache of the security broker and permissions token with a time stamp on the platform coordinator, Wobber further discloses a time-stamp associated with the token for limiting the validity of the token (see for example; col 6 ln 23-39). Authentication tokens refer to a data structure for use in authenticating a user. The record of Wobber (see for example col 6 ln 23-50) also refers to a data structure used for authenticating and authorizing users. Therefore, one of ordinary skill in the art at the time of the applicant's invention would have realized that such a record of Wobber correspond to the authentication token. Access credentials are well known in the art to vary on systems and by time through the policy set by a system administrator. The use of a time-stamp to limit the validity of tokens according to time would have improved security by making tokens that were valid before the new policy was created invalid and thus denying access to a potential unauthorized user. It would have been obvious for one of ordinary skill in the art

at the time of the applicant's invention to use the token associated with a time-stamp of Wobber within the Moriconi-Andrew combination because it would have added extra security by putting a time-limit on the validity of the token.

As for relaying the permissions token to the platform coordinator, Wobber further discloses relaying the token to a client for future authentication (see for example, col 7 ln 5-63). When using such tokens in the Moriconi-Andrews combination, one of ordinary skill in the art at the time of the applicant's invention would have realized the relaying of permissions token in place of the authentication response.

As per claim 12, Moriconi further discloses invoking programmatically, an embedded component within the software application when a secured resource is accessed (see for example col 11 ln 7-17); passing a resource name identifying the secured resource through the embedded component to a platform coordinator (see for example; col 8 ln 23-31 and col 10 ln 64-66); and relaying the access rights to the software application through the embedded component (see for example; col 8 ln 25-31 and col 7 ln 7-17). The relaying of access rights is inherent to pass authorization to the application requesting authorization credentials.

Andrews further discloses retrieving an authentication token from a platform coordinator (see for example; col 22 ln 1-24). Andrews further discloses

comparing the secured resource name with permissions tokens on the platform coordinator for a matching permission token (see for example; col 22 ln 10-54). Andrews discloses that each security call is mapped to a token and that the token contains user identification information and access credentials. Therefore, one of ordinary skill in the art at the time of the applicant's invention would realize that such comparison and determining of access rights for a user is essentially a comparison of tokens.

Furthermore, Moriconi-Andrews does not explicitly teach the tokens being on a cache. Wobber discloses the use of tokens on a cache (see for example col 7 ln 5-63) to expedite the access validation of a user gaining access to a resource (see for example; col 2 ln 5-23).

Moriconi-Andrew and Wobber disclose a means of user authentication and authorization to resources. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Wobber within the Moriconi-Andrew combination because it would have provided a means of granting authentication and access to resources in a quicker means through the use of caching tokens (see for example; Wobber, col 2 ln 5-23).

As for retrieving an authentication token from a cache on the platform coordinator, Moriconi discloses authenticating a user (see for example; col 4 ln 1-17). Wobber further discloses tokens (records) for authenticating valid users (see for example; col 6 ln 7-50). Furthermore, tokens are well known in the art serve as a means for ensuring authentication users before access to resources

are granted. It would have been obvious for one of ordinary skill in the art to use such an authentication token in the Moriconi-Andrew combination because it would have provided increased security through extra validation of an authenticated and authorized user of such a resource.

4. Claims 13-14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Moriconi et al (hereinafter Moriconi), US Patent 6,158,010.

As per claim 13, Moriconi discloses a method of providing computer security (see for example, abstract), comprising: securing a plurality of resources within a software application (see for example; col 6 ln 4-15); identifying secured resources within a software applications (see for example; col 6 ln 4-15); selecting some of the plurality of resources (see for example; col 6 ln 23-27 and col 9 ln 23-60; the resources are secured based on a security policy, therefore the plurality of resources are selected based on the security policy); grouping secured resources into user roles in a data store (see for example col 6 ln 64-col 7 ln 3); creating a plurality of user names (see for example; col 6 ln 64-73) and aliases in the data store (see for example; col 7 ln 12-25) and each alias being associated with the same one user role (see for example; user roles; col 6 ln 64-col 7 ln 5). When using user roles, user names must also be created to identify individual users to the role, such that a policy manager can manage the users in each user role (see for example; col 12 ln 53-62).

Moriconi further discloses determining access privileges to the plurality of resources using an alias corresponding to a user name by virtue of the same one user role from one of the plurality of data stores (see for example; col 8 ln 23-31). The alias has the same access rights (privileges) corresponding to the user (see for example; col 7 ln 13-24). The access rights are determined by the user role associated with the user (see for example col 7 ln 55-57), therefore the alias also corresponds to the user role of the user name.

As for replicating the plurality of resources, the user roles, the plurality of user names and the plurality of aliases in a plurality of data stores, Moriconi further discloses servers for maintaining users (see for example col 7 ln 5-8). One of ordinary skill in the art at the time of the applicant's invention would have realized the means of replicating in a plurality of data stores must be done in order to ensure proper synchronization between data stores.

As per claim 14, Moriconi further discloses authenticating the user as a valid user (see for example; col 4 ln 1-18), authorizing the user to access one of the secured resources in the software application (see for example; col 8 ln 23-32).

5. Claims 15-16 rejected under 35 U.S.C. 103(a) as being unpatentable over Moriconi et al (hereinafter Moriconi), US Patent 6,158,010, and further in view of Wu et al (hereinafter Wu), US Patent 5,774,551.

As per claim 15, Moriconi does not explicitly teach a specific authentication means. Wu discloses a means of authenticating a user comprising: retrieving a user identifier (see for example; col 17 ln 30-33); passing the user identifier to a security provider (see for example; col 17 ln 34-59); verifying the user identifier against a data store on the security provider (see for example; col 17 ln 50-59). As for returning an encrypted authentication token, Wu further discloses that the authentication tokens are encrypted (see for example; col 10 ln 63-65) and that user's authentication token are stored after users are authenticated (see for example; col 2 ln 23-25). The means of obtaining an authentication token for later use is well known in the art to be needed to establish a valid authentication token. One of ordinary skill in the art at the time of the applicant's invention would have realized the returning of authentication tokens for future use after the user is initially authorized. It would have been obvious for one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Wu within the system of Moriconi because it would have provided a means of authenticating a user in a quick and secure manner wherein subsequent verification can be expedited from the use of authentication tokens.

As per claim 16, Moriconi further discloses capturing a security call from the software application, the security call containing a name identifying a secured resource (see for example; col 8 ln 23-31); retrieving one of the plurality of

aliases from the data store of the security broker, the retrieved alias corresponding to the user identifier (see for example; col 8 ln 23-31). The alias corresponds to a user, thus corresponding to a user identifier (see for example; col 7 ln 12-25). Moriconi further discloses passing the retrieved alias to a security provider (see for example; col 10 ln 33-col 11 ln 5); and verifying the alias against a provider data store on the security provider (see for example; col 13 ln 14-32) to determine access rights to the secured resource (see for example col 13 ln 25-32).

Moriconi does not explicitly teach an encrypted permissions token. Wu discloses encrypted tokens for determination of access credentials (see for example; col 17 ln 35-59 and col 19 21-37) and that tokens contain a plurality of information pertaining to the user (see for example; col 2 ln 8-23). The use of tokens is well known in the art to ensure the validity of a user and for expediting user access time in subsequent access requests. It would have been obvious to one of ordinary skill the art at the time of the applicant's invention to combine the encrypted tokens of Wu within the system of Moriconi because it would have provided an increase in security through an extra layer of validity of a user and expedited subsequent requests to the same resource.

6. Claim 17 is rejected under 35 U.S.C. 103(a) as being unpatentable over Moriconi et al (hereinafter Moriconi), US Patent 6,158,010, and further in view of Wu et al

Art Unit: 2135

(hereinafter Wu), US Patent 5,774,551, and further in view of Kausik et al, US Patent 6,263,446.

As per claim 17, Moriconi further discloses gathering information about a user for authorizing access to secured resources, from the group consisting of a digital signature (see for example; col 4 ln 1-16), a user name and password (see for example; col 61-col 2 ln 11) and hardware token (see for example, smart cards, col 4 ln 1-16) Smart cards are well known in the art to be a form of a hardware token.

As for software tokens, Moriconi-Wu does not explicitly teach such a software token. Kausik discloses a means of authentication and authorizing users using a software token (see for example; software wallet, col 1 ln 45-55 and col 4 ln 1-17). Software tokens are well known to be a secure means of authorizing users to access credentials (see for example; Kausik, col 1 ln 45-55). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Kausik within the Moriconi-Wu combination because it would have provided a means of using secure and valid data in authorizing access to secured resources.

7. Claims 18-19, 21-24, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Boitana, US Patent 5,305,456, in view of Moriconi et al (hereinafter Moriconi), US Patent 6,158,010.

As per claim 18, Boitana discloses a computer security system comprising: a plurality of computers workstations, each computer workstation having an operating system and software application installed (see for example; col 10 ln 9-25 and fig 3A and fig 6), the software application containing an embedded component (see for example; application control component, col 5 ln 54-62); a plurality of security providers, each security provider having a security data store (see for example; col 6 ln 15-25). Security providers, such as RACF, are well known in the art to include data store to hold user access credentials for use in authorizing users a security broker, each security broker, having a data store, see for example (col 7 ln 67-col 8 ln 9), the security broker being a computer in network communication with the computer workstations and the security providers (see for example; Intermediate Security Transactions, fig 6), wherein each computer workstation is capable of communicating with each security broker; and wherein each security broker is capable of communicating with each security provider (see for example; fig 6).

Boitana does not explicitly teach a plurality of security brokers. Moriconi discloses a means of authorizing users through a plurality of security brokers (see for example; col 11 ln 7-17). It is well known in the art that increasing the amount of systems (security brokers) for processing data will result in increased performance and reliability due to the shared resources between the systems (security brokers). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Moriconi within

the system of Boitana because it would have provided greater performance and reliability of the system due to shared processing.

As per claim 19, Boitana-Moriconi discloses the claimed limitations as described above (see claim 18). As for a platform coordinator, Moraine further discloses an application control interface for routing permissions requests to security brokers (see for example; col 10 ln 41-51 and col 11 ln 6-17). Therefore, one of ordinary skill in the art at the time of the applicant's invention would have realized the platform coordinator to be present in such applications for the routing to be carried out. Furthermore, Moriconi discloses such a plurality of security brokers for the increase in performance and reliability (see for example; col 11 ln 7-17). The means of routing to other brokers for proceeding with authentication and authorization when one broker is unavailable is well known in the art to be part of the reliability increase for having such a plurality of brokers in Moriconi. Therefore, one of ordinary skill in the art at the time of the applicant's invention would have realized such a routing to establish the increase in reliability.

As per claim 21, Boitana-Moriconi discloses the claimed limitations as described above (see claim 18). Boitana further discloses routing permissions requests programmatically to the security providers (see for example; col 10 ln 56-63), each security provider being capable of routing permissions requests to any one of the security providers (see for example; col 10 ln 56-63). Boitana

does not explicitly teach the routing such that if one security provider is unavailable, the security broker can route permissions requests to another security provider. Moriconi further discloses the means of providing a plurality of processing units for increase in performance and reliability (see for example; col 11 ln 7-18). Moriconi discloses such a plurality of units for the increase in performance and reliability (see for example; col 11 ln 7-17). The means of routing to other brokers for proceeding with authentication and authorization when one broker is unavailable is well known in the art to be part of the reliability increase for having such a plurality of brokers in Moriconi. Therefore, one of ordinary skill in the art at the time of the applicant's invention would have realized such a routing to establish the increase in reliability. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to perform such a routing because it would have provided an increase in both performance and reliability of authenticating and authorizing users.

As per claim 22, Boitana-Moriconi discloses the limitations as described above (see claim 18). Boitana further discloses administration utilities for configuring, updating, maintain the data store and the security data store (see for example; col 4 ln 5-33). Boitana does not explicitly teach a single software application for maintaining user identifiers, setting and changing permissions, creating security events, and tracking system usage and security events within the security system. Moriconi discloses such a single software application (see

for example; col 9 ln 44-col 10 ln 7). The use of a single software application is well known in the art to add convenience to users, wherein they only need to run one application. It would have been obvious for one of ordinary skill in the art at the time of the applicant's invention to combine the single software application of Moriconi within the system of Boitana because it would have provided a convenient and central area for a system administrator to change and enforce different criteria of the security system.

As per claim 23, Boitana discloses a mean of authorizing access rights to secured resources in a software application comprising: authenticating a computer user to a computer security provider via a user identifier corresponding to the computer user (see for example; col 6 ln 4-25), the computer security provider returning a result to a security broker according to the user identifier (see for example; col 6 ln 18-25), storing the result on the security broker (see for example; col 6 ln 20-25). Boitana further discloses retrieving user information from the security broker (see for example col 6 ln 25-36) and computer security provider returning surrogate permissions to the security broker, the surrogate permissions corresponding to the user identifier (see for example; col 8 ln 45-65), the surrogate permissions for determining access rights to secured resources in the software application according to the surrogate permissions (see for example, col 8 ln 60-65).

Boitana does not explicitly teach retrieving a surrogate identifier from the security broker, the surrogate identifier corresponding the result, and the surrogate identifier being undisclosed to the computer user. Moriconi further discloses a surrogate identifier (see for example; col 7 ln 25-40). Each user is mapped to a global user. Furthermore, in computer programming, an identifier must be assigned to identify such a mapping for a user. One of ordinary skill in the art at the time of the applicant's invention would have realized the surrogate identifier needing to be present in order to identify the global user in the computer programming art. Both Boitana and Moriconi disclose a means of authenticating and authorizing a user to a secured resource. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to use such surrogate identifiers of Moriconi within the system of Boitana to provide an efficient means of identifying users across different platforms wherein each platform may have a user under a different identifier. Furthermore, Moriconi discloses the use of surrogate identifiers for use in authorizing the user to requested resources (see for example; col 8 ln 23-32) and that the surrogate identifier is to increase efficiency by centralizing management of users throughout the system (see for example; col 7 ln 25-33). One of ordinary skill in the art at the time of the applicant's invention would have been able to use the authentication process of Boitana as authentication of users to the system and further use the surrogate identifier in the authorizing steps of Boitana so as to relieve the system of processing users with similar privileges. In such a

combination, one of ordinary skill in the art at the time of the applicant's invention would have realized such retrieving and authorizing with surrogate identifiers in process of authorizing users.

As per claim 24, Boitana-Moriconi discloses the limitations as described above (see claim 23). Boitana further discloses passing the identifier to a security manager (see for example, col 6 ln 20-25); querying for the identifier in a permissions list on the security provider using the security manager (see for example; col 6 ln 20-25); determining surrogate permissions for the identifier according to the permissions list; and returning the surrogate permissions to the security broker (see for example; col 10 ln 41-55). The security providers, such as RACF, are well known in the art to incorporate a security manager to manage querying and determining of the result according to a permissions list (access credentials). Therefore, one of ordinary skill in the art at the time of the applicant's invention would have realized such a security manage and steps of authorizing with a security provider as being incorporated in the well known security providers of Boitana. Furthermore, Boitana discloses the authorizing of a user identifier. The surrogate identifier taught by Moriconi and the authorizing of surrogate identifiers to a security provider in place of user identifiers is described above (see claim 23).

As per claim 26, Boitana-Moriconi discloses the claimed limitations as described above (see claim 23). Boitana further discloses passing the identifier to a security manager (see for example, col 6 ln 20-25); querying for the identifier in a permissions list on the security provider using the security manager (see for example; col 6 ln 20-25); determining validity of the user identifier according to the authentication list; and returning the result to the security broker (see for example; col 10 ln 41-55). The security providers, such as RACF, are well known in the art to incorporate a security manager to manage querying and determining of the result according to a permissions list (access credentials). Therefore, one of ordinary skill in the art at the time of the applicant's invention would have realized such a security manage and steps of authorizing with a security provider as being incorporated in the well known security providers of Boitana.

8. Claims 20 and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Boitana, US Patent 5,305,456, in view of Moriconi et al (hereinafter Moriconi), US Patent 6,158,010, and further in view of Wobber et al (hereinafter Wobber), US Patent 5,235,642.

As per claim 20, Boitana-Moriconi discloses the claimed limitations as described above. Moriconi further discloses a surrogate identifier (see for example; col 7 ln 25-40). Each user is mapped to a global user. Furthermore, in computer programming, an identifier must be assigned to identify such a

mapping for a user. One of ordinary skill in the art at the time of the applicant's invention would have realized the surrogate identifier needing to be present in order to identify the global user in the computer programming art. Both Boitana and Moriconi disclose a means of authenticating and authorizing a user to a secured resource. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to use such surrogate identifiers of Moriconi within the system of Boitana to provide an efficient means of identifying users across different platforms wherein each platform may have a user under a different identifier.

Boitana-Moriconi does not explicitly teach authentication tokens to retrieve a surrogate identifier. Wobber discloses the use of tokens on a cache (see for example col 7 ln 5-63) to expedite the access validation of a user gaining access to a resource, wherein the token is used to access a user identifier (see for example; col 2 ln 5-23).

Moriconi-Andrew and Wobber disclose a means of user authentication and authorization to resources. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Wobber within the Moriconi-Andrew combination because it would have provided a means of granting authentication and access to resources in a quicker means through the use of caching tokens (see for example; Wobber, col 2 ln 5-23).

As per claim 25, Boitana-Moriconi discloses the claimed limitations as described above. Boitana further discloses passing the surrogate permissions

from the security broker to a platform coordinator (see for example; col 8 ln 45-59); relaying the surrogate permissions to an embedded component within the software application (see for example; col 8 ln 60-65); and interpreting the surrogate permission using to permit or deny access rights to the secured resource (see for example; col 8 ln 60-65 and col 10 ln 56-63; a platform coordinator must exist for such transmitting of access signals between different platforms).

As for a function within the software application, the function capable of interpreting the surrogate permissions, Boitana discloses means of interpreting the surrogate permissions in a software application (see for example col 8 ln 60-65). In order for such interpreting to be accomplished, one of ordinary skill in the art at the time of the applicant's invention would have realized that such a function is inherent to the teachings of Boitana for such interpreting.

As for storing the surrogate permissions with a time stamp in a cache on the platform coordinator, Boitana discloses storing the surrogate permissions (see for example, 8 ln 45-65 and col 10 ln 40-63). However, Boitana-Moriconi does not explicitly teach storing the surrogate permissions with a time stamp in a cache. Wobber discloses storing authorization data on a cache with a time-stamp (see for example; col 6 ln 23-50). Both Wobber and the Boitana-Moriconi combination disclose a means of user authentication and authorization to secured resources. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Wobber within

the Boitana-Moriconi combination because it would have would have provided a means of granting authentication and access to resources in a quicker means through the use of caching tokens (see for example; Wobber, col 2 ln 5-23).

Conclusion

9. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

US Patent 6,088,451, to He et al, discloses a means of authorization using a user alias in place of a user identifier.

US Patent 5,748,890, to Goldberg et al, discloses a means of user authorization using user roles.


Any inquiry concerning this communication or earlier communications from the examiner should be directed to Allen S. Wu whose telephone number is 703-305-0708. The examiner can normally be reached on Monday-Friday 9am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Vu can be reached on 703-305-4393. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Allen Wu
Patent Examiner
Art Unit 2135

ASW



KIM WU
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100